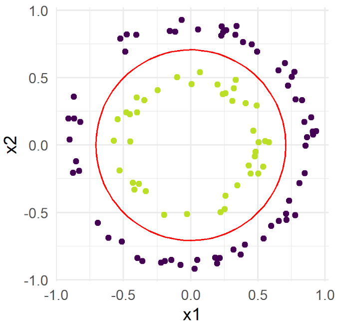
Introduction to Machine Learning

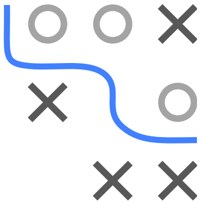
Feature Generation for Nonlinear Separation



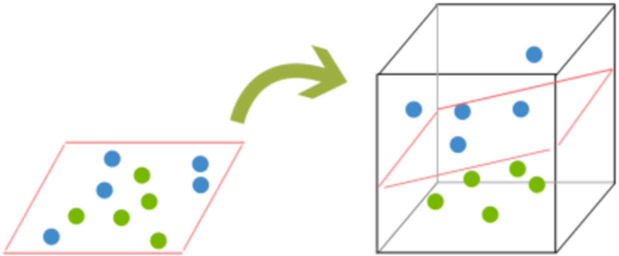
Learning goals

- Understand how nonlinearity can be introduced via feature maps in SVMs
- Know the limitation of feature maps

NONLINEARITY VIA FEATURE MAPS

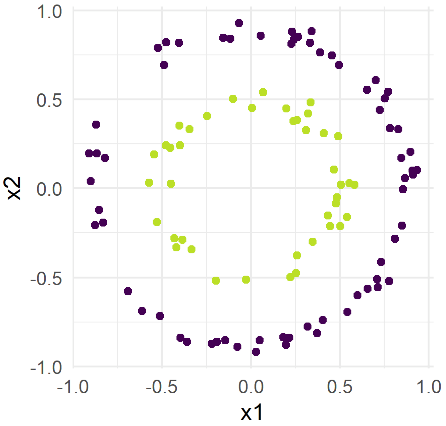


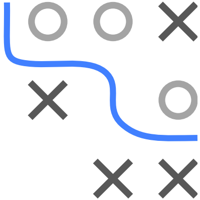
- How to extend a linear classifier, e.g. the SVM, to nonlinear separation between classes?
- We could project the data from 2D into a richer 3D feature space!



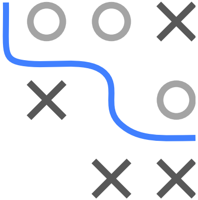
NONLINEARITY VIA FEATURE MAPS

In order to "lift" the data points into a higher dimension, we have to find a suitable **feature map** $\phi : \mathcal{X} \rightarrow \Phi$. Let us consider another example where the classes lie on two concentric circles:

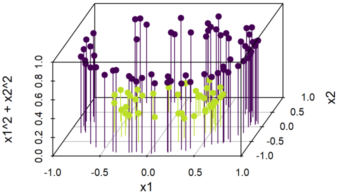


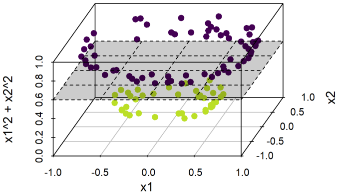


NONLINEARITY VIA FEATURE MAPS

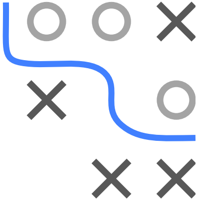


We apply the feature map $\phi(x_1, x_2) = (x_1, x_2, x_1^2 + x_2^2)$ to map our points into a 3D space. Now our data can be separated by a hyperplane.

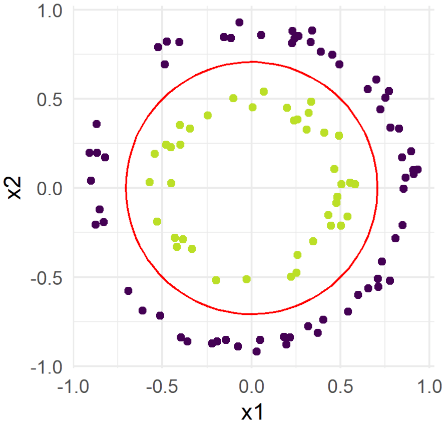




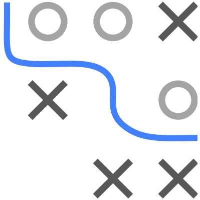
NONLINEARITY VIA FEATURE MAPS



The hyperplane learned in $\Phi \subset \mathbb{R}^3$ yields a nonlinear decision boundary when projected back to $\mathcal{X} = \mathbb{R}^2$.



FEATURE MAPS: COMPUTATIONAL LIMITATIONS



Let us have a look at a similar nonlinear feature map $\phi : \mathbb{R}^2 \to \mathbb{R}^5$, where we collect all monomial feature extractors up to degree 2 (pairwise interactions and quadratic effects):

$$\phi(x_1, x_2) = (x_1^2, x_2^2, x_1 x_2, x_1, x_2).$$

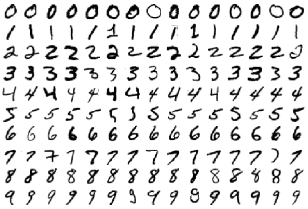
For p features vectors, there are k_1 different monomials where the degree is exactly d, and k_2 different monomials up to degree d.

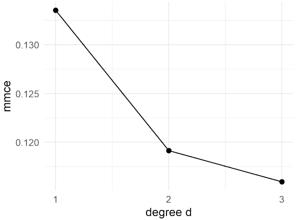
$$k_1 = \binom{d+p-1}{d} \qquad k_2 = \binom{d+p}{d} - 1$$

Which is quite a lot, if p is large.

FEATURE MAPS: COMPUTATIONAL LIMITATIONS

Let us see how well we can classify the 28×28-pixel images of the handwritten digits of the MNIST dataset (70K observations across 10 classes). We use SVM with a nonlinear feature map which projects the images to a space of all monomials up to the degree d and $C = 1$:



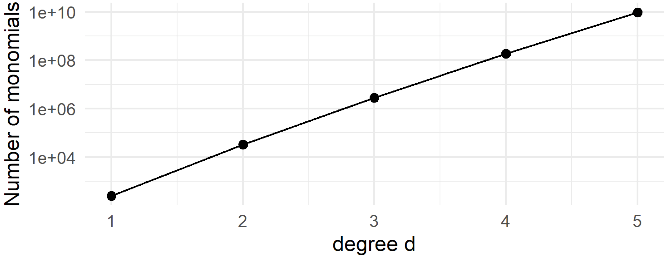


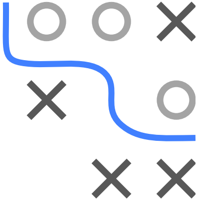
For this scenario, with increasing degree d the test mmce decreases.

NB: We handle the multiclass task with the "one-against-one" approach. We are somewhat lazy and only use 700 observations to train (rest for testing). We do not do any tuning - as we always should for the SVM!

FEATURE MAPS: COMPUTATIONAL LIMITATIONS

However, even a 16×16-pixel input image results in infeasible dimensions for our extracted features (monomials up to degree d).





In this case, training classifiers like a linear SVM via dataset transformations will incur serious **computational and memory problems**.

Are we at a “dead end”?
Answer: No, this is why kernels exist!